

WE CLAIM:

1. A method of executing a set of at least one incomplete task, comprising:
- 5 (a) selecting an incomplete task from the set;
- (b) resetting an execution timer having an expiry condition;
- (c) advancing execution of the selected task until the earlier of (i) completion of the selected
- 10 task and (ii) expiry of the execution timer; and
- (d) upon expiry of the execution timer prior to completion of the selected task, suspending execution of the selected task.
- 15 2. A method as defined in claim 1, further comprising:
- (e) returning to step (a) following suspension of the selected task.
- 20 3. A method as defined in claim 2, further comprising:
- (f) upon completion of the selected task prior to expiry of the execution timer, returning to step (a).
- 25 4. A method as claimed in claim 1, further comprising:
- (e) upon completion of the selected task prior to expiry of the execution timer, removing the selected task from the set and returning to step (a).
- 30 5. A method as defined in claim 1, wherein selecting an incomplete task from the set includes selecting an

incomplete task on the basis of how recently that task has become a member of the set.

6. A method as defined in claim 1, wherein selecting an incomplete task from the set includes selecting an incomplete task on the basis of an expected duration for that task.

7. A method as defined in claim 1, wherein selecting an incomplete task from the set includes selecting an incomplete task on the basis of a priority associated with that task.

8. A method as defined in claim 1, wherein selecting an incomplete task from the set includes selecting an incomplete task on the basis of the number of times that the task has been previously suspended.

9. A method as defined in claim 1, wherein selecting an incomplete task from the set includes selecting an incomplete task on the basis of how long ago the task was first suspended.

10. A method as defined in claim 1, wherein selecting an incomplete task from the set includes selecting an incomplete task on the basis of a random function.

11. A method as defined in claim 1, wherein advancing execution of the selected task includes beginning the selected task if the selected task has not been previously suspended.

12. A method as defined in claim 1, wherein advancing execution of the selected task includes resuming the selected task if the selected task has been previously suspended.

5

13. A method as defined in claim 11, wherein advancing execution of the selected task includes resuming the selected task if the selected task has been previously suspended.

10

14. A method as defined in claim 13, wherein suspending the selected task includes saving a context associated with the selected task.

15. A method as defined in claim 14, wherein resuming the selected task includes retrieving the previously saved context associated with the selected task.

16. A method as defined in claim 15, wherein the context associated with the selected task includes variables local to the selected task.

17. A method as defined in claim 15, wherein the context associated with the selected task includes a state of the selected task.

18. A method as defined in claim 15, wherein the context associated with the selected task includes a state of a central processing unit (CPU).

30

19. A method as defined in claim 1, wherein the expiry condition of the execution timer is a pre-determined number of clock cycles.

5 20. A method as defined in claim 1, wherein the expiry condition of the execution timer is a pre-determined period of time.

10 21. A method as defined in claim 1, wherein the expiry condition of the execution timer is a pre-determined percentage of completeness of the selected task.

22. A method as defined in claim 1, further comprising:
if the selected task is a new version of an existing
15 task in the set for which execution is more advanced than for the selected task, removing the existing task from the set.

23. A method as defined in claim 1, wherein suspending
20 the selected task includes saving a context associated with the selected task.

24. A method as defined in claim 23, wherein the context associated with the selected task includes variables
25 local to the selected task.

25. A method as defined in claim 23, wherein the context associated with the selected task includes a state of the selected task.

30

26. A method as defined in claim 23, wherein the context associated with the selected task includes a state of a central processing unit (CPU).

5 27. A method as defined in claim 1, further comprising:
adding at least one incomplete task to the task set
in response to receipt of a message.

10 28. Computer-readable media tangibly embodying a program
of instructions executable by a computer to perform the
method of claim 1.

29. An apparatus, comprising:
a memory for storing a task set;
15 a task scheduler for adding incomplete tasks to the
task set; and
a task processor for executing incomplete tasks in
the task set by selecting an incomplete task from the
set; resetting an execution timer having an expiry
20 condition; advancing execution of the selected task until
the earlier of (i) completion of the selected task and
(ii) expiry of the execution timer; and upon expiry of
the execution timer prior to completion of the selected
task, suspending execution of the selected task.

25

30. A task processor for executing a set of at least one
incomplete task, comprising:

- (a) means for selecting an incomplete task from the
set;
30 (b) means for resetting an execution timer having
an expiry condition;

(c) means for advancing execution of the selected task until the earlier of (i) completion of the selected task and (ii) expiry of the execution timer; and

5 (d) means for suspending execution of the selected task upon expiry of the execution timer prior to completion of the selected task.

31. A computer readable storage medium containing a
10 program element for execution by a computing device to implement a task processing entity, the program element including:

(a) program code means for selecting an incomplete task from the set;

15 (b) program code means for resetting an execution timer having an expiry condition;

(c) program code means for advancing execution of the selected task until the earlier of (i) completion of the selected task and (ii) expiry of the execution timer; and
20

(d) program code means for suspending execution of the selected task upon expiry of the execution timer prior to completion of the selected task.